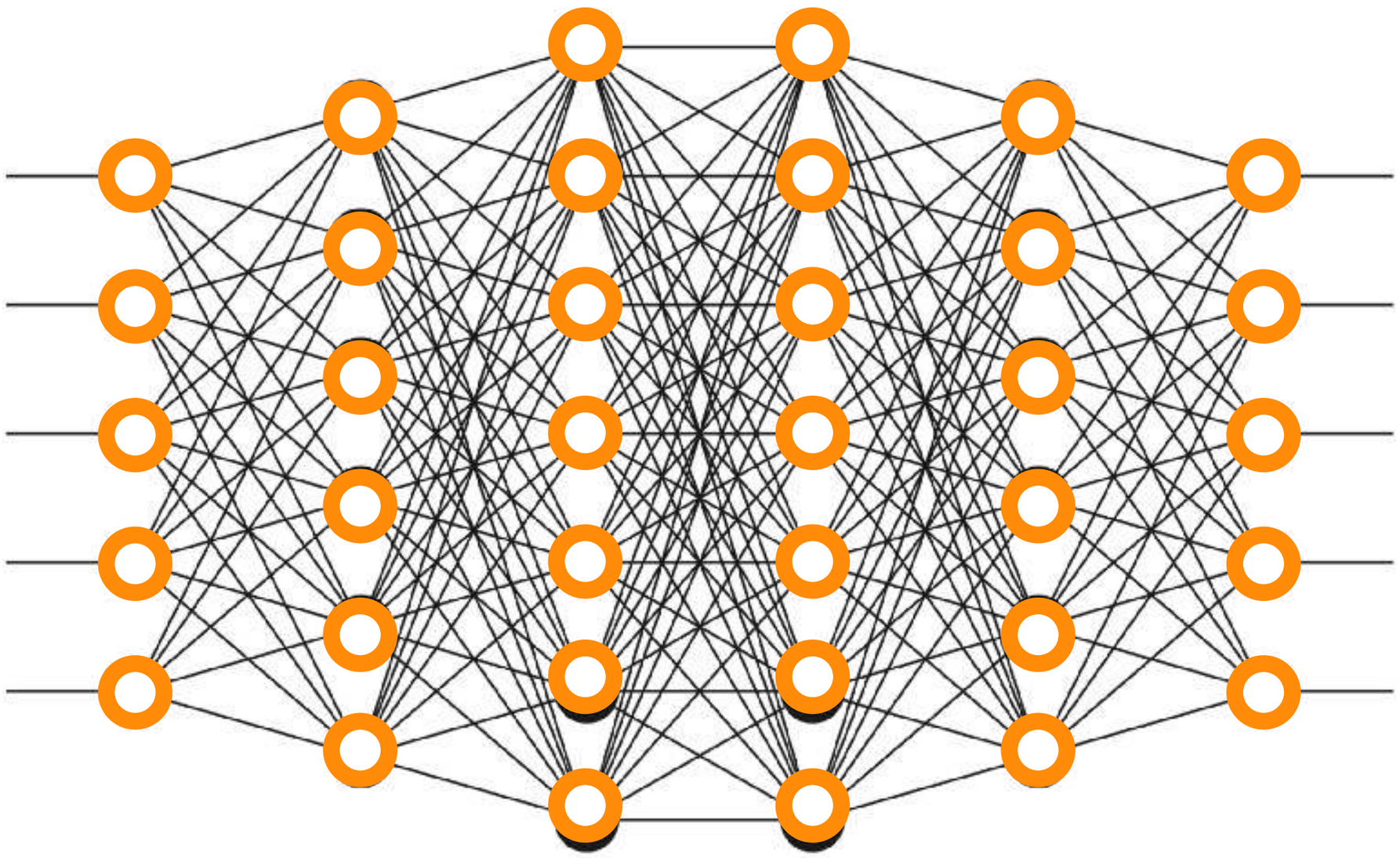
BOSSCODER ACADEMY

LEARN

DEEP LEARNING

in 6 Weeks



DEEP LEARNING

Disclaimer

Mastering deep learning in just 6 weeks is an ambitious goal.

This plan aims to provide a **solid foundation** and equip you with the **basic skills** to continue learning and build upon.

WEEK 1

Deep Learning Fundamentals

Topics

- Introduction to Deep Learning: Definition, Applications
- Artificial Neural Networks: Structure, Activation Functions, Loss Functions
- Gradient Descent Optimization: Basic Principles, Learning Rate
- Introduction to Python Libraries: NumPy, Pandas, Matplotlib (for basic understanding)

Self-Test Questions

- Explain the difference between supervised and unsupervised learning.
- Describe the role of an activation function in a neural network.
- What is the purpose of gradient descent optimization?
- Briefly explain the functionalities of NumPy and Pandas.

Project

- Implement a simple Perceptron algorithm for binary classification using Python libraries.

WEEK 2

Feedforward Neural Networks

Topics

- Multi-layer Perceptrons: Architecture, Backpropagation Algorithm
- Common Activation Functions: Sigmoid, ReLU, Softmax
- Introduction to Overfitting and Regularization Techniques

Self-Test Questions

- Multi-layer Perceptrons: Architecture, Backpropagation Algorithm
- Common Activation Functions: Sigmoid, ReLU, Softmax
- Introduction to Overfitting and Regularization Techniques

Project

- Build a multi-layer perceptron model from scratch (without libraries) to classify handwritten digits using the MNIST dataset.

WEEK 3

Convolutional Neural Networks (CNNs)

Topics

- Introduction to CNNs: Convolutional layers, Pooling layers, Filters
- Applications of CNNs: Image Recognition, Object Detection
- Introduction to Hyperparameter Tuning: Learning rate, number of filters, etc.

Self-Test Questions

- Describe the functionality of a convolutional layer in a CNN.
- Explain the different types of pooling operations used in CNNs.
- Why is hyperparameter tuning crucial in deep learning models?

Project

- Implement a simple CNN architecture (without using libraries) to classify images of different categories (e.g., cats vs. dogs).

WEEK 4

Recurrent Neural Networks (RNNs)

Topics

- Understanding Sequence Data: Time Series, Text, Natural Language Processing (NLP)
- Introduction to RNNs: Vanilla RNN, Long Short-Term Memory (LSTM) networks
- Applications of RNNs: Machine translation, sentiment analysis, music generation

Self-Test Questions

- Differentiate between traditional neural networks and RNNs.
- Explain the vanishing gradient problem in RNNs and how LSTMs address it.
- Describe two applications of RNNs in the field of NLP.

Project

- Implement a simple RNN model (without using libraries) to predict the next word in a sequence, given a starting sentence.

WEEK 5

Generative Adversarial Networks (GANs)

Topics

- **Focus:** Deep dive into Generative Adversarial Networks (GANs)
- Understanding the architecture and training process of GANs
- Applications of GANs: Image generation, data augmentation, style transfer
- Introduction to Ethical Considerations in Deep Learning: Bias, Fairness, Explainability

Self-Test Questions

- Explain the concept of a Generative Adversarial Network (GAN).
- Describe the two main components of a GAN and their roles.
- Discuss one ethical concern related to the use of GANs and potential mitigation strategies.

WEEK 6

Deep Learning Project Exploration

Work on a Project:

1. **Choose a project:** Consider your interests, skill level, and available resources (data, computing power).
2. **Define the problem:** Clearly articulate what your project aims to achieve and the specific task it will perform.
3. **Gather data:** Find or create a suitable dataset for your chosen task. Ensure data quality and relevance.
4. **Select and experiment with models:** Choose a suitable deep learning model for your project (e.g., CNN for image recognition, RNN for text processing). Experiment with different architectures and hyperparameters (learning rate, batch size, etc.) to optimize performance.
5. **Evaluate and improve:** Evaluate your model's performance using relevant metrics (accuracy, precision, recall, etc.). Identify areas for improvement and iterate on your model and training process.
6. **Document and present your work:** Document your project's journey, including data sources, model architecture, code, and evaluation results. Prepare a presentation or report summarizing your findings.

Project Ideas:

- **Image Classification:** Extend your CNN from Week 3 to a more complex dataset (e.g., CIFAR-100, Fashion MNIST) with more categories.
- **Time Series Forecasting:** Train an RNN or LSTM model to predict stock prices, weather patterns, or website traffic.
- **Chatbot Development:** Build a simple chatbot using natural language processing techniques like sentiment analysis and sequence-to-sequence learning.
- **Music Generation:** Train an RNN model on a music dataset to generate new melodies or complete existing ones.
- **Object Detection in Images:** Utilize pre-trained models like YOLO or SSD to detect objects in images or videos.
- **Anomaly Detection:** Train a model to identify unusual patterns in data, such as fraudulent transactions or equipment failure.

BOSSCODER ACADEMY

WHY BOSSCODER?

1000+ Alumni placed at Top Product-based companies.

More than **136% hike** for every **2 out of 3** working professional.

Average package of **24LPA.**

The syllabus is most **up-to-date** and the list of problems provided covers **all important topics.**

Lavanya
Meta

Course is very well **structured** and **streamlined** to crack any **MAANG** company

Rahul .
Google

EXPLORE MORE